

STORE

Structured Mode Syntax

```
STORE [RECORD] [IN] [FILE] view-name
      [PASSWORD = operand1]
      [CIPHER = operand2]
      [ [USING]
        [GIVING] NUMBER operand3 ] [(r)]
```

Reporting Mode Syntax

```
STORE [RECORD] [IN] [FILE] view-name
      [PASSWORD = operand1]
      [CIPHER = operand2]
      [ [USING]
        [GIVING] NUMBER operand3 ]
      { [USING] SAME [RECORD] [AS] [STATEMENT [(r)]]
        [SET
        WITH] [operand4 = operand5]... }
```

Operand	Possible Structure					Possible Formats												Referencing Permitted	Dynamic Definition
Operand1	C	S				A												yes	no
Operand2	C	S					N											yes	no
Operand3		S					N	P										no	yes
Operand4		S	A			A	N	P	I	F	B	D	T	L				no	no
Operand5	C	S	A			A	N	P	I	F	B	D	T	L				yes	no

Related Statements: UPDATE | DELETE | END TRANSACTION | BACKOUT TRANSACTION

Function

The STORE statement is used to add a record to a database.

Notes for DL/I databases:

This statement may be used to add a segment occurrence.

If the dataset is defined with a primary key, a value for the primary key field must be provided.

In the case of a GSAM database, records must be added at the end of the database (due to GSAM restrictions).

Note for SQL databases:

This statement may be used to add a row to a table. The PASSWORD, CIPHER, and GIVING NUMBER clauses cannot be used. The STORE statement corresponds with the SQL statement INSERT.

Note for VSAM databases:

If the dataset is defined with a primary key, a value for the primary key field must be provided.

view-name

As *view-name*, you specify the name of a view, which must have been defined either in a DEFINE DATA statement or outside the program in a global or local data area.

In reporting mode - if no DEFINE DATA statement is used - *view-name* may also be the name of a DDM.

PASSWORD/CIPHER

These clauses are applicable only for an Adabas or VSAM database.

The PASSWORD clause is used to provide a password when updating data from a file which is password-protected.

The CIPHER clause is used to provide a cipher key when updating data from a file which is enciphered.

See the statements FIND and PASSW for further information.

USING/GIVING NUMBER

This clause can only be used for an Adabas or VSAM database.

This option is used to store a record with a user-supplied Adabas ISN. If a record with the specified ISN already exists, an error message will be returned and the execution of the program will be terminated unless ON ERROR processing was specified.

Note for VSAM databases:

This clause is only valid for VSAM RRDS, in which case a user-supplied RRN (relative record number) corresponds to the ISN as described above.

SET/WITH

SET/WITH can be used in reporting mode to specify the fields for which values are being provided. Any field defined in the file that is not specified in the SET clause will contain a null value in the new record.

This clause is not permitted if a DEFINE DATA statement is used, because in that case the STORE statement always refers to the entire view as defined in the DEFINE DATA statement.

DL/I Considerations

Values must be provided for the segment sequence field, and for all sequence fields of the ancestors.

Only I/O (sensitive) fields may be provided.

A segment of variable length is stored with the minimum length necessary to contain all fields as specified with the STORE statement. The segment length will never be less than the minimum size specified in the SEGM macro of the DBD.

If a multiple-value field or a periodic group is defined as variable in length, at the end of the segment only the occurrences as specified in the STORE statement are written to the segment and define the segment length.

USING SAME

USING SAME can be used in reporting mode to indicate that the same field values as read in the statement referenced by the STORE statement (FIND, GET, READ) are to be used to add a new record.

This clause is not permitted if a DEFINE DATA statement is used, because in that case the STORE statement always refers to the entire view as defined in the DEFINE DATA statement.

System Variable *ISN

The Natural system variable *ISN contains the Adabas ISN or VSAM RBA/RRN respectively assigned to the new record as a result of the STORE statement execution. A subsequent reference to *ISN must include the statement number of the related STORE statement.

For VSAM databases, *ISN is available only for ESDS and RRDS files.

*ISN is not available for DL/I and SQL databases.

Example

```

/* EXAMPLE 'STOEX1': STORE
*****
RESET #BIRTH-D (D)
*
REPEAT
  INPUT 'ENTER A PERSONNEL ID AND NAME (OR ''END'' TO END)' //
    'PERSONNEL-ID : ' #PERSONNEL-ID (A8) //
    'NAME          : ' #NAME          (A20) /
    'FIRST-NAME    : ' #FIRST-NAME    (A15)
*****
*  VALIDATE ENTERED DATA
*
  IF #PERSONNEL-ID = 'END' OR #NAME = 'END' THEN STOP
  IF #NAME = ' ' THEN
    REINPUT WITH TEXT 'ENTER A LAST-NAME' MARK 2 AND SOUND ALARM
  IF #FIRST-NAME = ' ' THEN
    REINPUT WITH TEXT 'ENTER A FIRST-NAME' MARK 3 AND SOUND ALARM
*****
*  ENSURE PERSON IS NOT ALREADY ON FILE
*
  FIND NUMBER EMPLOYEES WITH PERSONNEL-ID = #PERSONNEL-ID
  IF *NUMBER > 0 THEN
    REINPUT 'PERSON WITH SAME PERSONNEL-ID ALREADY EXISTS'
    MARK 1 AND SOUND ALARM
*****
*  GET FURTHER INFORMATION
*
  INPUT
    'ADDITIONAL PERSONNEL DATA'          ////
    'PERSONNEL-ID          : ' #PERSONNEL-ID (AD=IO) /
    'NAME                  : ' #NAME          (AD=IO) /
    'FIRST-NAME            : ' #FIRST-NAME    (AD=IO) ///
    'MARITAL STATUS        : ' #MAR-STAT (A1)      /
    'DATE OF BIRTH (YYYYMMDD) : ' #BIRTH      (A8)      /
    'CITY                  : ' #CITY          (A20)      /
    'COUNTRY (3 CHARACTERS) : ' #COUNTRY      (A3)      //
    'ADD THIS RECORD (Y/N)  : ' #CONF        (A1) (AD=M)
*****
*  ENSURE REQUIRED FIELDS CONTAIN VALID DATA
*
  IF NOT (#MAR-STAT = 'S' OR = 'M' OR = 'D' OR = 'W')
    REINPUT TEXT 'ENTER VALID MARITAL STATUS S=SINGLE ' -
      'M=MARRIED D=DIVORCED W=WIDOWED' MARK 1
  IF NOT (#BIRTH = MASK(YYYYMMDD) AND #BIRTH = MASK(1582-2699))
    REINPUT TEXT 'ENTER CORRECT DATE' MARK 2
  IF #CITY = ' '
    REINPUT TEXT 'ENTER A CITY NAME' MARK 3
  IF #COUNTRY = ' '
    REINPUT TEXT 'ENTER A COUNTRY CODE' MARK 4
  IF NOT (#CONF = 'N' OR = 'Y')
    REINPUT TEXT 'ENTER Y (YES) OR N (NO)' MARK 5
  IF #CONF = 'N'
    ESCAPE TOP
*****
* continued on next page

```

```
* continued from previous page
*****
*   ADD THE RECORD
*
  MOVE EDITED #BIRTH TO #BIRTH-D (EM=YYYYMMDD)
*
  STORE RECORD IN EMPLOYEES
  WITH  PERSONNEL-ID = #PERSONNEL-ID
        NAME         = #NAME
        FIRST-NAME    = #FIRST-NAME
        MAR-STAT      = #MAR-STAT
        BIRTH         = #BIRTH-D
        CITY          = #CITY
        COUNTRY       = #COUNTRY
  END OF TRANSACTION
*
  WRITE NOTITLE 'RECORD HAS BEEN ADDED'
  RESET INITIAL #CONF
*
LOOP
END
```

```
ENTER A PERSONNEL ID AND NAME (OR 'END' TO END)

PERSONNEL-ID : 90001100

NAME         : JONES
FIRST-NAME    : EDWARD
```

```
ADDITIONAL PERSONNEL DATA

PERSONNEL-ID      : 90001100
NAME              : JONES
FIRST-NAME        : EDWARD

MARITAL STATUS    : m
DATE OF BIRTH (YYYYMMDD) : 690511
CITY              : wan chai
COUNTRY (3 CHARACTERS) : hkg

ADD THIS RECORD (Y/N) : y
```